**MONISH KUMAR N – 732219EC059**

**ASSIGNMENT - 2**

1, 2: - Create user table with EMAIL, USERNAME ,ROLL\_NO, PASSWORD and perform insert ,update and delete.

QUERY:

create table user1(username varchar(255),email varchar(255), roll\_no varchar(255),password varchar(255));

insert into user1 values('Raguram','raguram@gmail.com','76','sffgh@01'),('Madav','madavan@gmail.com','56','Zxfm@02'),('Sridhar','sridhar@gmail.com','87','Qwrty@07'),('Monishkumar','monishkumar@gmail.com','59','monishkumar@gmail.com');

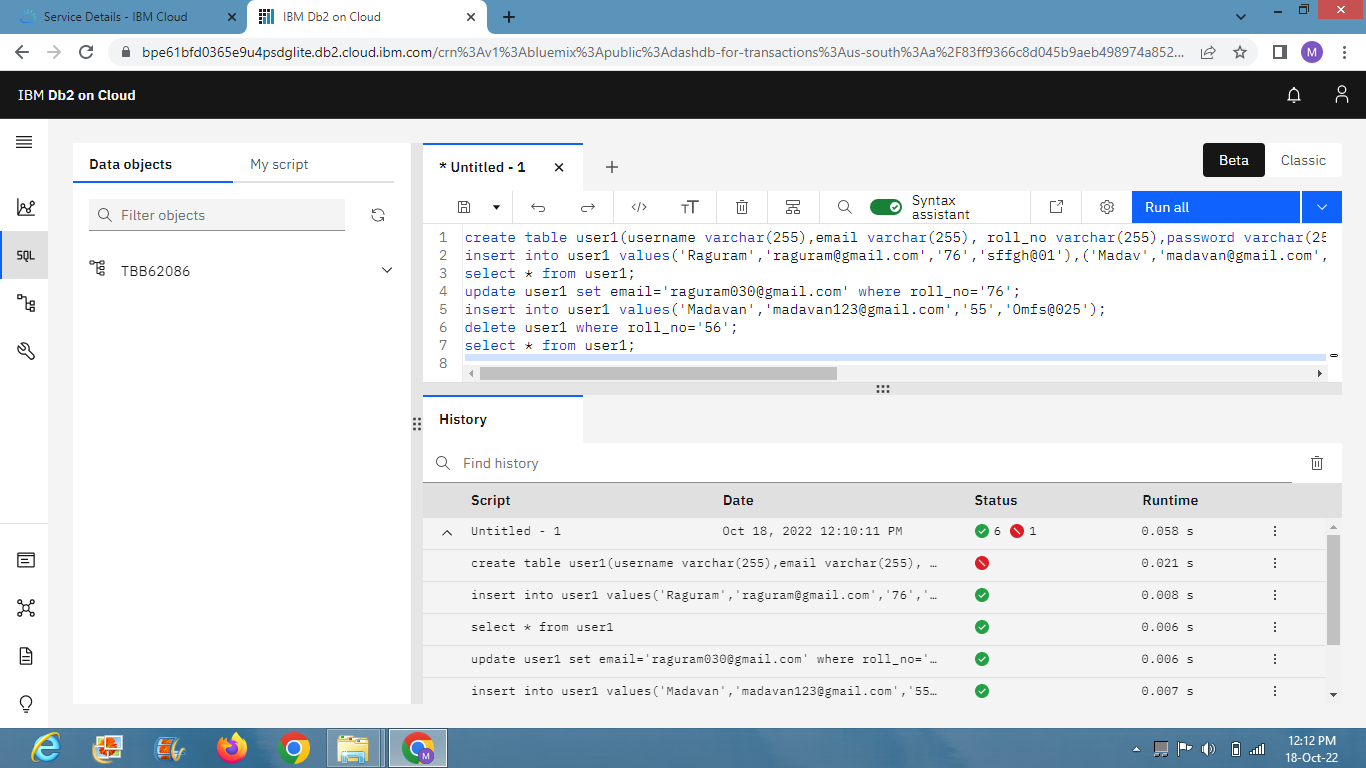
select \* from user1;

update user1 set email='raguram030@gmail.com' where roll\_no='76';

insert into user1 values('Madavan','madavan123@gmail.com','55','Omfs@025');

delete user1 where roll\_no='56';

select \* from user1;



3. Connect python to db2

from flask import Flask, render\_template, request, redirect, url\_for, session

import ibm\_db

import db

import db2

import re

hostname = 'fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud'

uid = 'tbb62086'

pwd = 'hZEDhhSNJxOQV86h'

driver = "{IBM DB2 ODBC DRIVER}"

db\_name = 'Bludb'

port = '32731'

protocol = 'TCPIP'

dsn = (

        "DATABASE ={0};"

        "HOSTNAME ={1};"

        "PORT ={2};"

        "UID ={3};"

        "SECURITY=SSL;"

        "PROTOCOL={4};"

        "PWD ={5};"

).format(db\_name, hostname, port, uid, protocol, pwd)

connection = ibm\_db.connect(dsn, "", "")

print(dsn)

try:

    print("Connecting to db2.....")

    db2 = ibm\_db.connect(dsn, "", "")

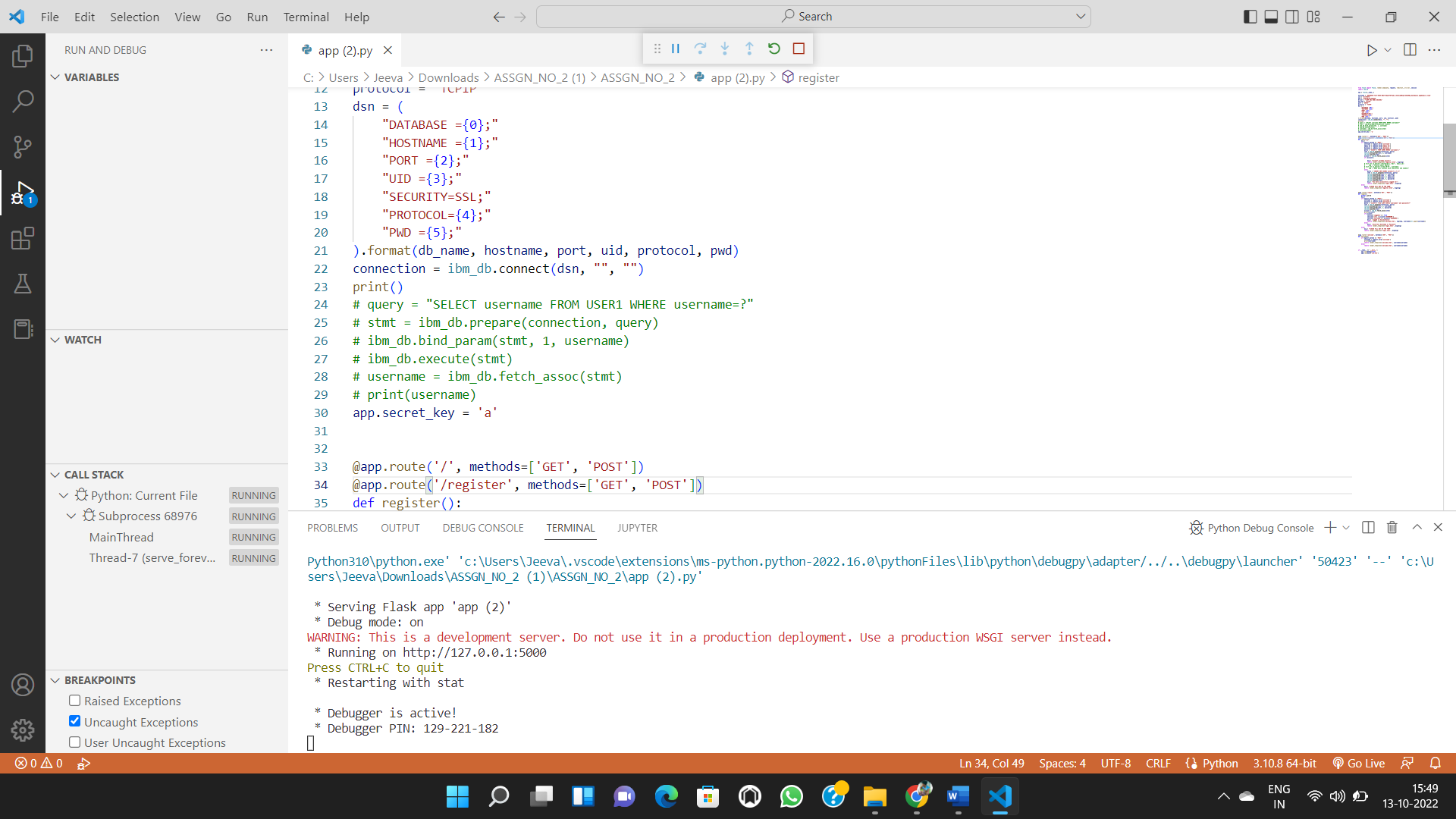
    print()

    print("Connected to database")

    print("Connection Successful!!!")

except Exception as exception:

    print("unable to connect ", exception)



**4) ACCESS LOGIN WITH CONNTING TO DATABASE**

from flask import Flask, render\_template, request, redirect, url\_for, session

import ibm\_db

app = Flask(\_\_name\_\_)

hostname = 'fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud'

uid = 'tbb62086'

pwd = 'hZEDhhSNJxOQV86h'

driver = "{IBM DB2 ODBC DRIVER}"

db\_name = 'Bludb'

port = '32731'

protocol = 'TCPIP'

dsn = (

    "DATABASE ={0};"

    "HOSTNAME ={1};"

    "PORT ={2};"

    "UID ={3};"

    "SECURITY=SSL;"

    "PROTOCOL={4};"

    "PWD ={5};"

).format(db\_name, hostname, port, uid, protocol, pwd)

connection = ibm\_db.connect(dsn, "", "")

print()

# query = "SELECT username FROM USER1 WHERE username=?"

# stmt = ibm\_db.prepare(connection, query)

# ibm\_db.bind\_param(stmt, 1, username)

# ibm\_db.execute(stmt)

# username = ibm\_db.fetch\_assoc(stmt)

# print(username)

app.secret\_key = 'a'

@app.route('/', methods=['GET', 'POST'])

@app.route('/register', methods=['GET', 'POST'])

def register():

    msg = " "

    if request.method == 'POST':

        username = request.form['username']

        email\_id = request.form['email\_id']

        phone\_no = request.form['phone\_no']

        password = request.form['password']

        query = "SELECT \* FROM USER1 WHERE username=?;"

        stmt = ibm\_db.prepare(connection, query)

        ibm\_db.bind\_param(stmt, 1, username)

        ibm\_db.execute(stmt)

        account = ibm\_db.fetch\_assoc(stmt)

        if (account):

            msg = "Account already exists!"

            return render\_template('register.html', msg=msg)

        # elif not re.match(r'[^@]+@[^@]+\.[^@]+', email\_id):

        #     msg = "Invalid email addres"

        # elif not re.match(r'[A-Za-z0-9+', username):

        #     msg = "Name must contain only characters and numbers"

        else:

            query = "INSERT INTO USER1 values(?,?,?,?)"

            stmt = ibm\_db.prepare(connection, query)

            ibm\_db.bind\_param(stmt, 1, username)

            ibm\_db.bind\_param(stmt, 2, email\_id)

            ibm\_db.bind\_param(stmt, 3, phone\_no)

            ibm\_db.bind\_param(stmt, 4, password)

            ibm\_db.execute(stmt)

            msg = 'You have successfully Logged In!!'

            return render\_template('login.html', msg=msg)

    else:

        msg = 'PLEASE FILL OUT OF THE FORM'

        return render\_template('register.html', msg=msg)

@app.route('/login', methods=['GET', 'POST'])

def login():

    global userid

    msg = ' '

    if request.method == "POST":

        username = request.form['username']

        password = request.form['password']

        query = "select \* from user1 where username=? and password=?"

        stmt = ibm\_db.prepare(connection, query)

        ibm\_db.bind\_param(stmt, 1, username)

        ibm\_db.bind\_param(stmt, 2, password)

        ibm\_db.execute(stmt)

        account = ibm\_db.fetch\_assoc(stmt)

        print(account)

        if account:

            session['Loggedin'] = True

            session['id'] = account['USERNAME']

            session['username'] = account['USERNAME']

            msg = 'Logged in Successfully'

            return render\_template('welcome.html', msg=msg, username=str.upper(username))

        else:

            msg = 'Incorrect Username or Password'

            return render\_template('login.html', msg=msg)

    else:

        msg = 'PLEASE FILL OUT OF THE FORM'

        return render\_template('login.html', msg=msg)

@app.route('/welcome', methods=['GET', 'POST'])

def welcome():

    if request.method == 'POST':

        username = request.form['username']

        print(username)

        return render\_template('welcome.html', username=username)

    else:

        return render\_template('welcome.html', username=username)

if \_\_name\_\_ == "\_\_main\_\_":

    app.run(debug=True)

    app.run(host='0.0.0.0')